

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A method of generating an input file using a meta language regarding graphics data compression, the method comprising:

making an extensible markup language (XML) schema that defines at least a compression node describing object data to be compressed, and parameters used to for data compression;

making style sheets which support conversion of an input XML file into a file, which is to be input to a data compression encoder, based on the XML schema; and

generating a the file, which is to be input to the data compression encoder, by parsing the input XML file based on the XML schema and the style sheets.

2. (Currently Amended) The method of claim 1, wherein the XML schema further ~~comprises at least~~ defines EncodingHints specifying the location of a file in which the object data to be compressed is stored.

3. (Original) The method of claim 1, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

4. (Original) The method of claim 2, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

5. (Currently Amended) A method of generating an input file using a meta language regarding graphics data compression, the method comprising:

making an XMT schema which defines a compression node~~[[,]]~~ ~~which defines~~ describing object data to be compressed, parameters for data compression, and BitWrapperEncodingHints which at least specifies ~~the~~ a location of a file in which the object data to be compressed is stored;

making an XMT2BIFS style sheet which supports conversion of an XMT input file into ~~an~~ a scene file and an XMT2MUX style sheet which supports conversion of the XMT input file into a mux file, based on the XMT schema; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets, respectively.

6. (Currently Amended) The method of claim 5, wherein the compression node comprises:

a node field ~~containing~~ comprising the object data to be compressed;

~~a buffer field which is not used together with an url field at the same time and~~
temporarily stores a bitstream defined in the compression node using an in-band scenario; and

~~the url~~ a URL field which ~~is not used together with the buffer field at the same time and~~ links information regarding the bitstream defined in the compression node using an out-band scenario,

wherein either the buffer field or the URL field is used.

7. (Currently Amended) The method of claim 6, wherein the compression node further comprises a type field specifying ~~the~~ a type of node compression scheme.

8. (Original) The method of claim 5, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object to be compressed, and a parameter for three-dimensional (3D) mesh information to be compressed.

9. (Currently Amended) The method of claim 5, wherein the BitWrapperEncodingHints further specifies an object descriptor ID that is the same as a URL ID[[,]] of the compression node, ~~the~~ a name of a file transmitting a compressed bitstream, and ~~the~~ a type of a stream format, wherein the file name ~~being~~ is described in the mux file.

10. (Currently Amended) The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file describing the compression node that defines the original data, compression parameters, and buffer; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets, respectively,

wherein the scene file comprises the ~~original~~ object data to be compressed, the data compression parameters, and a buffer which temporarily stores a bitstream obtained from compression of the original data, and

the mux file describes ~~the~~ a name of a file obtained by encoding the scene file using a BIFS encoder and a stream format.

11. (Currently Amended) The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file defining the compression node ~~containing the~~ comprising a buffer temporarily storing the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets, respectively.

wherein the scene file ~~contains~~ comprises the buffer temporarily storing ~~the~~ a bitstream that is a representation of the compressed object data, and

the mux file specifies ~~the~~ a name of a file obtained by encoding the scene file using ~~the~~ a BIFS encoder, and ~~the~~ a stream format.

12. (Currently Amended) The method of claim 5, wherein parsing the input XMT file comprises:

receiving the input XMT file which describes the compression node defining the ~~original~~ object data to be compressed, the data compression parameters, and ~~url~~ URL information, and the BitWrapperEncodingHints defining an object descriptor ID that is the same as a ~~url~~ URL ID of the compression node and ~~the~~ a location of ~~the~~ a bitstream that is a representation of the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets, respectively.

wherein the scene file ~~contains~~ comprises the ~~original~~ object data to be compressed, the data compression parameters, and ~~url~~ the URL information which links information regarding ~~the~~ a bitstream obtained from compression of the ~~original~~ object data, and

the mux file specifies the location of the bitstream, which is a representation of the compressed object data, and ~~the~~ a stream format defined in the BitWrapperEncodingHints.

13. (Currently Amended) The method of claim 12, wherein the input XMT file further comprises an ObjectDescriptorUpdate which defines an object descriptor ID that is the same as the object descriptor ID specified in the BitWrapperEncodingHints, and ~~the~~ a name of ~~a~~ the mux file to be generated from the parsing of the input XMT file,

wherein the scene file further specifies ~~an~~ the object descriptor ID that is the same as the object descriptor ID specified in the BitWrapperEncodingHints, and the name of the mux file.

14. (Currently Amended) The method of claim 5, wherein parsing the input XMT file further comprises:

receiving the input XMT file describing the BitWrapperEncodingHints specifying the compression node defining ~~the url~~ a URL which links information regarding already-compressed object data, an object descriptor ID that is the same as ~~the url~~ a URL ID, and ~~the~~ a location of ~~the~~ a bitstream that is a representation of the compressed object data; and

generating the scene file and the mux file by parsing the input XMT file using the XMT schema and the XMT2BIFS and XMT2MUX style sheets, respectively.

wherein the scene file ~~containing~~ comprises the URL which links information regarding the bitstream obtained from the compression of the ~~original~~ object data and whose ID is the same as the ID of the object descriptor specified in the compression node, and

the mux file specifies the location of the bitstream that is a representation of the compressed object data defined in the BitWrapperEncodingHints and ~~the~~ a stream format.

15. (Currently Amended) The method of claim 14, wherein the input XMT file further comprises an ObjectDescriptorUpdate which specifies an object descriptor ID that is the same as the object descriptor ID defined in the BitWrapperEncodingHints and ~~the~~ a name of the mux file obtained by the parsing of the input XMT file,

wherein the scene file further specifies ~~an~~ the object descriptor ID that is the same as the object descriptor ID defined in the BitWrapperEncodingHints and the name of the mux file.

16. (Original) A computer readable recording medium for recording a program executing the method of claim 1 in a computer.

17. (Original) A computer readable recording medium for recording a program executing the method of claim 5 in a computer.

18. (Currently Amended) A system for generating an input file using a meta language regarding graphics data compression, the system comprising:

an XML schema that defines a compression node ~~at least containing~~ comprising at least information regarding object data to be compressed and parameters used for data compression;

style sheets which support conversion of ~~the~~ an input XML file into a file which is to be input to a predetermined data compression encoder, based on the XML schema; and

an XML parser which parses the input XML file based on the XML schema and the style sheets to generate a the file which is to be input to a predetermined data compression encoder.

19. (Original) The system of claim 18, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object, and a parameter for three-dimensional (3D) mesh information to be compressed.

20. (Currently Amended) A system for generating an input file using a meta language regarding graphics data compression, the system comprising:

an XMT schema which defines a compression node specifying object data to be compressed, parameters for data compression, and BitWrapperEncodingHints which at least specifies ~~the~~ a location of a file in which the object data to be compressed is stored;

an XMT2BIFS style sheet which supports conversion of an input XMT file into a scene file based on the XMT schema;

an XMT2MUX style sheet which supports conversion of the input XMT file into a mux file based on the XMT schema; and

an XMT parser which parses the input XMT file based on the XMT schema and the XMT2BIFS and XMT2MUX style sheets to generate the scene and mux files, respectively, as the input files to a predetermined compression encoder.

21. (Currently Amended) The system of claim 20, wherein the compression node comprises:

a node field specifying the object data to be compressed;

~~a buffer field which is not used together with a url field at the same time and~~ transmits a bitstream that is a representation of the compressed object data using the an in-band scenario, the bitstream being specified in the compression node; and

~~the url~~ a URL field which ~~is not used together with the buffer field at the same time and~~ transmits the bitstream that is a representation of the compressed object data using the an out-band scenario, the bitstream being specified in the compression node,

wherein the compression node uses either the buffer field or the URL field.

22. (Original) The system of claim 20, wherein the parameters comprise at least one of a parameter for keyframe-based animation data regarding vertex coordinates of the object, a parameter for rotation information of the object, a parameter for position information of the object, and a parameter for three-dimensional (3D) mesh information to be compressed.

23. (Currently Amended) The system of claim 20, wherein the BitWrapperEncodingHints further comprises an object descriptor ID that is the same as an a URL ID defined in the compression node, the a name of a file transmitting the a compressed bitstream contained in the mux file, and the a type of the a stream format.

24. (New) A method of generating an input file using meta representation of graphics data compression, the method comprising:

making an extensible MPEG-4 textual format (XMT) schema defining a compression node which includes information regarding object data to be compressed, an encoding parameter required for data compression, objectDescriptorID which is the same as identification stored in a URL field of the compression node, and BitWrapperEncodingHints which specifies a name of a file storing a compressed bitstream to be transmitted and a format of the bitstream, the file being included in a mux file;

generating, based on the XMT schema, an XMT2BIFS style sheet which supports conversion of an input XMT file into a scene file and an XMT2MUX style sheet which supports conversion of the input XMT file into the mux file,

wherein the scene file includes UPDATE OD describing the objectDescriptorID which is the same as the identification stored in the URL field of the compression node and a name of the mux file, when graphics data, which is to be compressed, is present or the URL of the compression node is used, and

wherein the mux file includes:

InitialObjectDescriptor having ES_Descriptor which is an elementary stream descriptor, ES_Descriptor including ES_ID which is identification of an elementary stream representing a bitstream obtained by compressing scene data, decConfigDescr specifying a decoder to be used to decode the bitstream, slConfigDescr indicating a reproduction time, and muxInfo indicating a name of file storing the bitstream to be transmitted and a format of the bitstream, and

ObjectDescriptor having ES_Descriptor which is an elementary stream descriptor, ES_Descriptor including objectDescriptorID which is identification of a URL linked to a bitstream of an object to be transmitted to the URL when the URL of the compression node is used, decConfigDescr specifying a decoder to be used to decode the bitstream, slConfigDescr indicating a reproduction time, and muxInfo indicating a name of file storing the bitstream to be transmitted and a format of the bitstream; and

making the scene file and the mux file by parsing the input XMT file according to the XMT schema, using the XMT2BIFS style sheet and the XMT2MUX style sheet,

wherein the compression node includes:

- a node field containing the object data to be compressed,
- a type field indicating a type of a node compression encoder,
- a URL field which is not used at the same time as a buffer field and transmits the bitstream defined in the compression node as out-band data, and
- the buffer field which is not used at the same time as the URL field and transmits the bitstream defined in the compression node as in-band data, and

wherein the encoding parameter comprises:

a parameter for keyframe-based animation data regarding vertex coordinates of the object to be compressed,

a parameter for three-dimensional mesh information,

a parameter for rotational moving keyframe-based animation data, and

a parameter for positional moving keyframe-based animation data.

25. (New) The method of claim 24, wherein the parsing of the input XMT file comprises:

receiving the XMT input file describing a compression node which includes a node field which contains original data, an encoding parameter, and a buffer; and

making a scene file and a mux file by parsing the input XMT file according to the XMT schema, using the XMT2BIFS and XMT2MUX style sheets,

wherein the scene file includes a compression node having a node field containing the original data, an encoding parameter, and type information specifying a buffer transmitting a bitstream obtained by compressing the original data and a type of an encoder, and

wherein the mux file includes a name of a file obtained by encoding the scene file using a BIFS encoder, and a format of the compressed bitstream.

26. (New) The method of claim 24, wherein the parsing of the input XMT file comprises:

receiving the input XMT file defining a compression node containing information regarding a buffer temporarily storing already compressed object data; and

making a scene file and a mux file by parsing the input XMT file according to the XMT schema, using the XMT2BIF style sheet and the XMT2MUX style sheet,

wherein the scene file includes a compression node having a node field which does not contain object data, and type information specifying a buffer transmitting a bitstream obtained by compressing the object data and a type of an encoder, and

wherein the mux file includes a name of a file obtained by encoding the scene file using a BIFS encoder, and stream format information.

27. (New) The method of claim 24, wherein the parsing of the input XMT file comprises:

receiving the input XMT file defining a compression node which describes a node field containing original data, an encoding parameter, and identification stored in a URL; objectDescriptorID which is the same as identification stored in a URL field of the compression node; and ObjectDescriptorUpdate having BitWrapperEncodingHints indicating a name of a file storing a compressed bitstream, which is included in the mux file and is to be transmitted, and a format of the bitstream; and

making a scene file and the mux file by parsing the input XMT file according to the XMT schema, using the XMT2BIFS style sheet and the XMT2MUX style sheet,

wherein the scene file includes:

a compression node having a node field containing the original data, an encoding parameter, and type information containing information regarding a URL linked to information regarding a bitstream obtained by compressing the original data and a type of an encoder, and

UPDATE OD indicating objectDescriptorID which is the same as identification stored in a URL field of the compression node, and a name of the mux file, and

wherein the mux file includes a name of a file which stores the compressed bitstream, which is included in the mux file defined in BitWrapperEncodingHints, and is to be transmitted, and information of a format of the compressed bitstream.

28. (New) The method of claim 1, wherein the parsing of the input XMT file comprises:

receiving the input XMT file defining a compression node describing a node field which contains already compressed object data, and a URL linked to the already compressed object data; objectDescriptorID which is the same as identification stored in a URL field of the compression node; and ObjectDescriptorUpdate including BitWrapperEncodingHints which indicates a name of a file storing a compressed bitstream, which is defined in a mux file and is to be transmitted, and a format of the compressed bitstream; and

making a scene file and the mux file by parsing the input XMT file according to the XMT schema, using the XMT2BIFS style sheet and the XMT2MUX style sheet,

wherein the scene file includes:

a compression node having a node field which does not contain the already compressed object data, and type information specifying identification stored in a URL field linked to information regarding a bitstream of the already compressed object data, and a type of an encoder, and

UPDATE OD having objectDescriptorID which is the same as identification stored in a URL field of the compression node, and a name of the mux file, and

wherein the mux file includes a name of a file storing the compressed bitstream which is included in the mux file defined in BitWrapperEncodingHints and is to be transmitted, and information of a format of the compressed bitstream.